www.icgst.com

Precise Obstacles Avoidance System for Visually Impaired People using Xbox 360 Kinect

D. Barrinuevo, G. Jacosalem, J. N. Lagahit, M. J. Lobedica, M. R. Salar, and R. Tolentino
Polytechnic University of the Philippines – Santa Rosa Campus, Tagapo, Laguna, Philippines
danicabarrinuevo@yahoo.com, gladysgeleraj@gmail.com, ljneil.jn@gmail.com, joylobedica@yahoo.com, rebeccasalar17@gmail.com, kenmetara@yahoo.com
http://www.pup.edu.ph

Abstract
This study proposes a system for obstacle avoidance for visually impaired people that uses Canny Edge Detector to eliminate the reliance of detection in the segmentation of the floor plane. The system acquires the images through Microsoft Kinect of Xbox 360 which is the primary input device. The process occurs in the depth image that will undergo depth thresholding to limit the distance. The resulting image from depth thresholding will undergo noise reduction to fill the broken areas. Then the edge detection through Canny will be performed. The extracted edges that appear as a contour will be used to determine the presence of an obstacle and will be enclosed in a bounding box drawn to the color image. Hereafter, decision making will be made which is responsible for determining if the user must go forward, left, right or stop through a sound feedback. The system was tested based on the rate of its detection out of 385 samples and on a real-time trial navigation on a structured environment by a blindfolded and blind user.

Keywords: *Canny Edge Detector, visually impaired, obstacle detection, Kinect.*

1. Introduction
About 285 million people around the world are estimated to be visually impaired, of whom 39 million are totally blind and 246 million have low vision [1]. The sense of sight is an outstanding feature that enables one to access and perceive the environment that surrounds them. One of the most difficult activities that must be conducted by these individuals is independent mobility which relates to sensing the obstacles and potential paths in the vicinity for the purpose of navigating through it. To help them navigate safely, without colliding any obstacles, several mobility and navigational aids were made. Some of these tools were the white cane and the guide dogs.
Obstacle detection is considered one of the most important tasks for a navigation system. It is responsible for determining and locating the presence of an obstacle on a considered region to avoid probable collision for safe travel. A number of sensors have been used for this purpose, including ultrasonic sensors [2] [3], laser range finders, and cameras. Recently, researchers have developed vision-based systems for obstacle detection from low-cost depth camera. Some existing studies utilizes Microsoft Kinect to detect and calculate the distance between the obstacle and the user [4] and some finds obstacles by detecting the corners in the RGB image and the input from the depth sensor provides the corresponding distance from Kinect's infrared sensor [5]. Apparently, some studies lack the capability of detecting the obstacle based on its substantial boundary. The extraction of obstacle's boundary, which will enable the exploitation of shape information, is important in identifying objects in images and segmenting images into individual objects.
Another method for support system to detect obstacle in indoor environment based on Kinect sensor and 3D image processing for the visually impaired person implements the Point Cloud Library (PCL) for data acquisition and Random Sample Consensus (RANSAC) algorithm for plane segmentation in the point cloud data [6]. This enables the detection of near obstacles such as walls, doors, stairs, and loose obstacles on the floor in order to assist the visually impaired people in their mobility. However, the reliance in floor segmentation affects the detection. In case that the floor ground detection failed, the system will not be able to detect the obstacles and if an obstacle involves a large horizontal plane, the obstacle could be mistakenly identified as the ground plane. The researchers proposed a system that uses Canny Edge detector which aims to improve current obstacle detection systems for the visually impaired persons by eliminating the dependence of detection process in the floor segmentation.

2. Method
The authors of this study conducted the implementation and evaluation of the system through a laptop, running on an Intel(R) Core(TM) i5-6200U CPU. The succeeding

block diagram shows the flow of the concepts that constitute the proposed system in Figure 1.

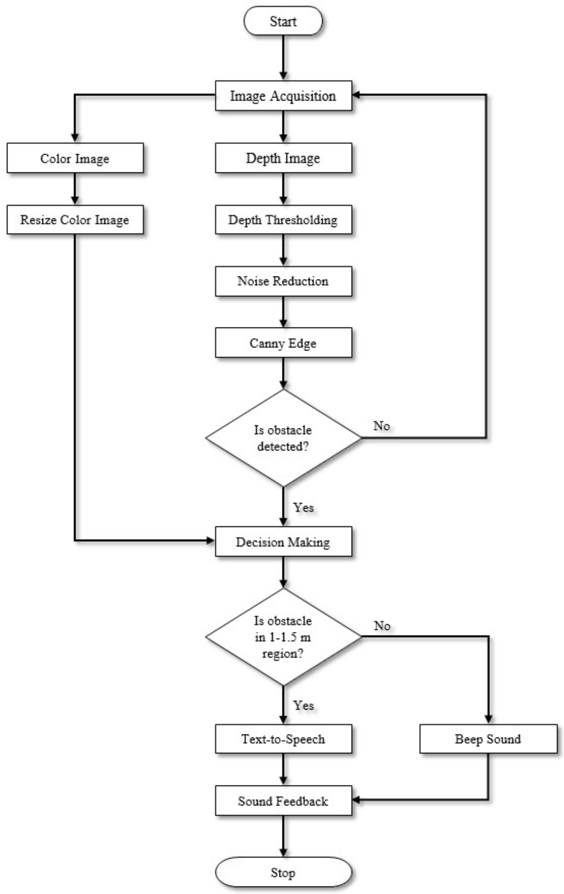

Figure 1. Framework of the proposed system

The system, in general, is composed of three main parts: the Kinect sensor, the laptop and the headphone. Firstly, the Kinect sensor transmits the stream of captured raw data to the system. Secondly, the laptop which contains the software that will be used for the processing, transforms the acquired raw data into significant information comprising of the reliable decision in the form of a sound feedback. Finally, the sound feedback will be conveyed through the use of the headphone in order to communicate the generated decision to the user. Figure 2 shows the setup of the system.

Figure 2. Hardware setup of the system

Image Acquisition

The system begins by capturing the raw data from the scene and converting it to a more suitable representation through image acquisition. Image acquisition is the first stage of any computer vision system. It provides the source image that will be used for processing.

The physical limits of Kinect to measure depth information under default mode is within 800 mm to 4000 mm, with horizontal and vertical angle of vision are 57.50 and 43.50, respectively. These physical limits of the sensor are considered in the development of the system.

Depth Thresholding

Depth thresholding is used to disregard distances that are situated outside the considered region. It is accomplished by setting the pixels with corresponding depth within the maximum and minimum threshold to white (255), otherwise, it will be set to black (0) as shown in Figure 3.

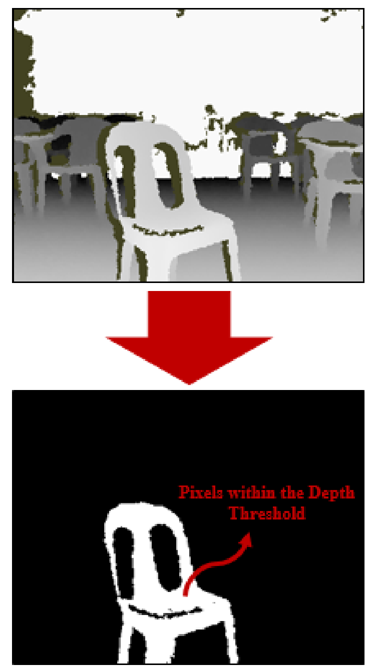

Figure 3. The image before and after passing through depth thresholding

The minimum reliable distance of 800 mm of the sensor is used as the minimum threshold and the maximum threshold of 1500 mm is selected which is based on the walking speed of visually impaired, that is 0.4 m/s when the presence of an obstacle is sensed [7].

Noise Reduction

Due to the hardware limitations of Kinect, the depth image can be broken producing black spots where no depth information is acquired. Noise reduction is used to patch up the broken areas in the image to make it more complete. The system used closing to reduce the noise encountered. A closing, which uses a rectangular 4x4 structuring element, was employed to fill the broken black areas of the depth image. Figure 4 illustrates the image that has undergone noise reduction.

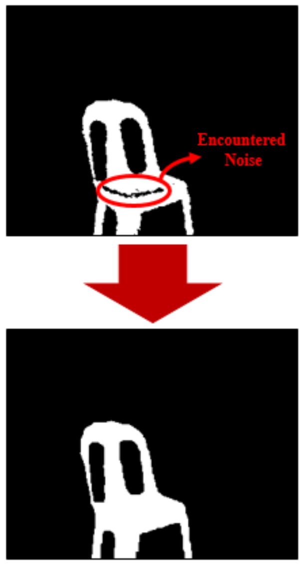

Figure 4. The image after passing through noise reduction

Canny Edge Detector

Edge detection is a method of finding an edge in an image [8]. The authors were led to the utilization of an edge detection algorithm which enables the detection of the actual shape of the obstacle and will resolve the case of misdetection due to plane segmentation. Edge detection is performed by a variety of ways. These techniques were grouped into two categories: the gradient based and laplacian based. Specific examples of these edge detection techniques include Sobel operator, Robert's cross operator, Prewitt's operator, Laplacian of Gaussian and Canny edge detection algorithm. Among other edge detection techniques, Canny was selected due to its low error rate, edge localization and its response to single edge which is substantial in determining the actual position of the obstacle in the image. Besides, among all edge detection techniques, Canny Edge detection algorithm is found to perform better under almost all circumstances [9].

Canny edge detection algorithm is a detection operator that uses a multi-stage algorithm to detect a wide range of edges in images. It is performed in this study to acquire the pixel location of the contour that constitutes the boundary of the obstacle. The process of Canny can be broken down to 5 different steps.

The first step is to apply Gaussian filter to smooth the image in order to remove the noise. This optimizes the trade-off between noise filtering and edge localization. Then, computing the magnitude of gradients in the image by using a 2x2 filter will be done. After that, non-maximum suppression is used to get rid of spurious response to edge detection to create thin lines. Thick lines may represent edges that are not in the location of the actual edge. Thus non-maximum suppression can help to suppress all the gradient values which indicate location with the sharpest change of intensity value. Then, double thresholding is applied to determine potential edges. It is used to reduce false edges. Last is the tracking of edges through hysteresis. This is used to link edges by suppressing all the other edges that are weak and not connected to strong edges.

The proposed system used Canny edge to extract the outline that depicts the boundary of the obstacle as shown in Figure 5.

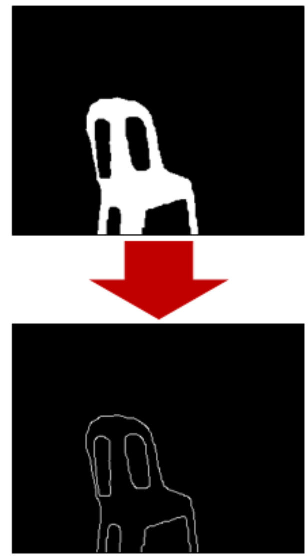

Figure 5. Result of applying Canny Edge Detection

Decision Making

Decision making is responsible for initiating the user to go forward, left, right or stop. It is done, first, by eliminating contours whose area is lower than the set limit to extract the real contours of obstacles. Then, a bounding box in the resized color image is used to surround the extracted obstacle. Based from observations, a minimum value for the area was set by the authors. Using the pixel location of the remaining contours, a bounding box can be plotted in the resized color image as shown in Figure 6.

Figure 6. Formation of the bounding box

Then, the image will be divided into 3 regions: the left, the center, and the right. Figure 7 shows the illustration used in partitioning the size of the areas in the divided image.

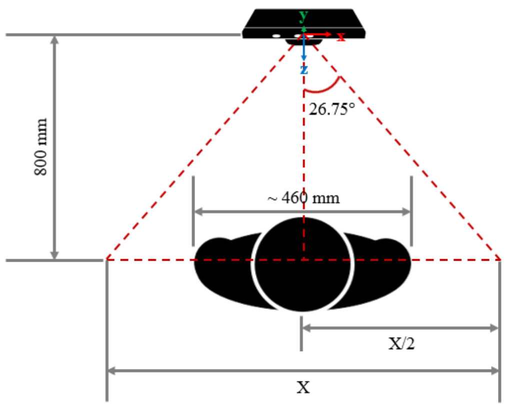

Figure 7. The diagram used to determine the partition of the area

The average measurement of the human figure based on real world measurements, which is approximately 460 mm, will be used to match the width covered by the center area. Due to the fact that the area covered in the image is wider at smaller distances, the minimum depth threshold of 800 mm is used to determine the proportions of the three areas in the divided image. In order to get the real size captured by the frame at a depth of 800 mm, this formula is used:

$$X = 2(\textit{minimum depth threshold}) \tan 26.75^{\circ} \qquad (1)$$

Knowing the real measurement of the frame and the average width of the human figure, as shown in Figure 8, the width (W) of the center area in pixels can be determined by using proportionality.

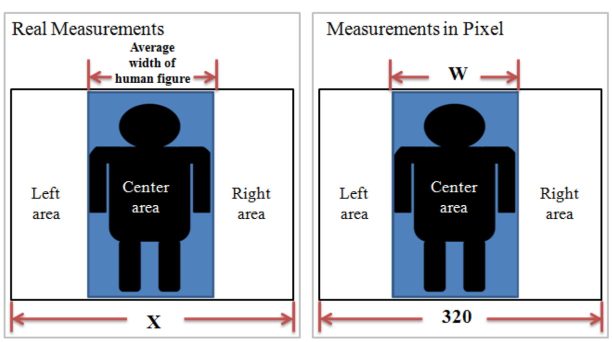

Figure 8. The dimensions in terms of pixel and real measurements needed in decision making

This can be quantified as the equation:

$$W = \frac{(320(\textit{average width of human figure}))}{X} \qquad (2)$$

And to identify the width of the right and left area, the formula used is:

$$W_{\textit{right area}} = W_{\textit{left area}} = \frac{W_{\textit{total area}} - W_{\textit{center area}}}{2W_{\textit{total area}}} \qquad (3)$$

Thus, the division of the area is shown in Figure 9.

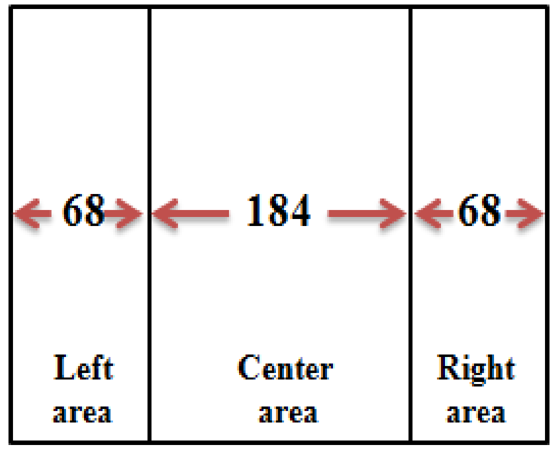

Figure 9. The division of the frame and the width in pixel of each area

The decision is based on the occurrence of the obstacle in the divided captured area of the processed image. The decision is made when an obstacle is detected 1000 mm from the user and following the conditions shown in Table-1.

Table-1. The decision for the following condition

Condition	Decision
If there is no obstacle in the center area.	Forward
If there is an obstacle in the center area but the right area is free.	Right
If there is an obstacle in the center area but the left area is free.	Left
If there is no area free of obstacle.	Stop

The system will give commands to the user if he must go forward, right, left, or stop depending upon where the obstacle was found. Figure 10 to Figure 13 demonstrates the system response in the case of a forward decision cited in the application window.

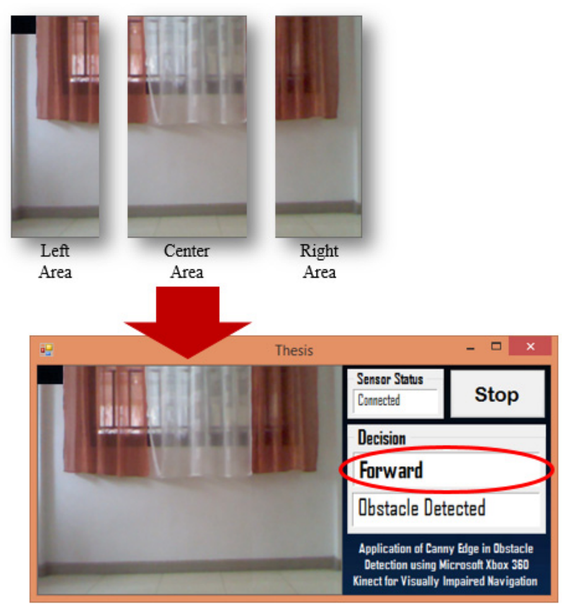

Figure 10. Forward decision

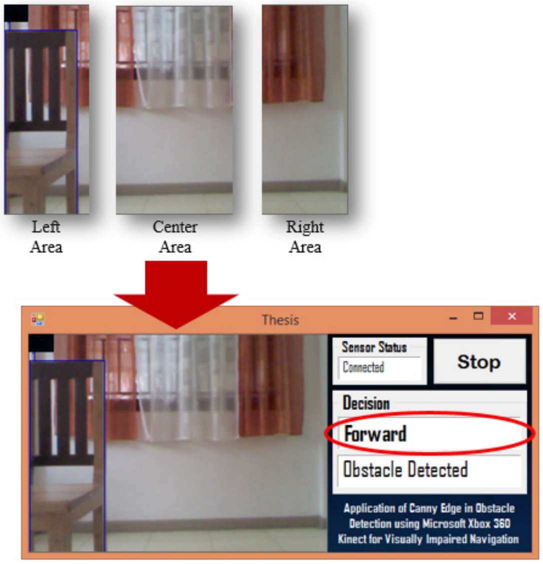

Figure 11. Forward decision

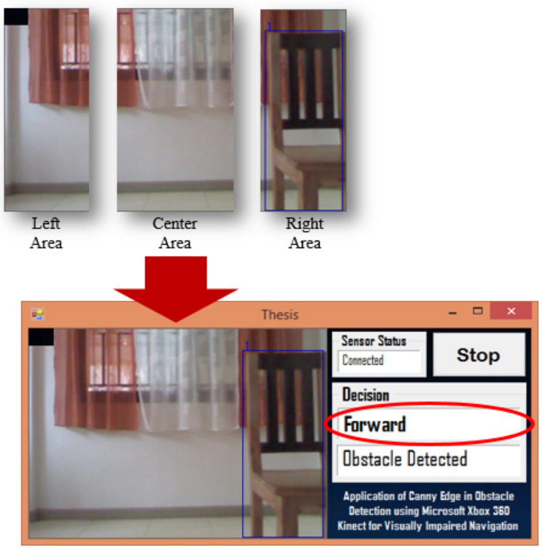

Figure 12. Forward decision

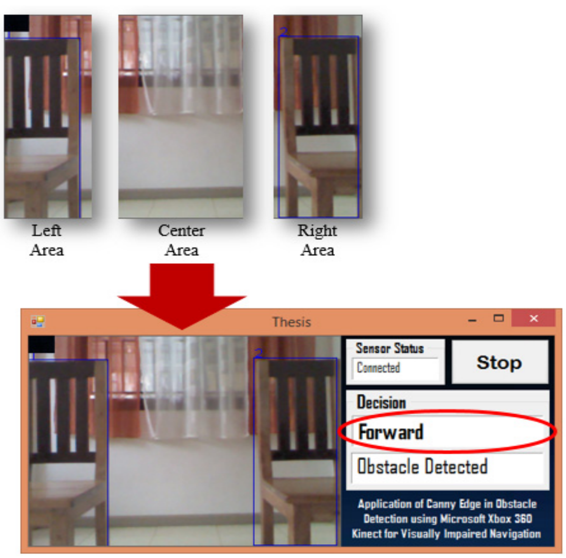

Figure 13. Forward decision

In the previous image, the obstacle was not found in the center area, then the system commanded the user with "FORWARD". The same command was also given when the left and right area is occupied by an obstacle. The system will tell the user to turn "RIGHT" if the center area is occupied and the right area is free from obstacle. This condition is illustrated in Figure 14.

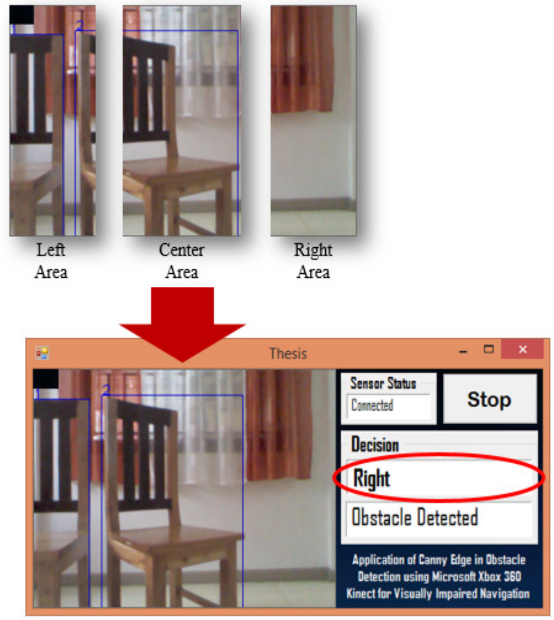

Figure 14. Right decision

Figure 15 demonstrates a "LEFT" decision delivered since the right and center area were occupied while the left area was safe to take.

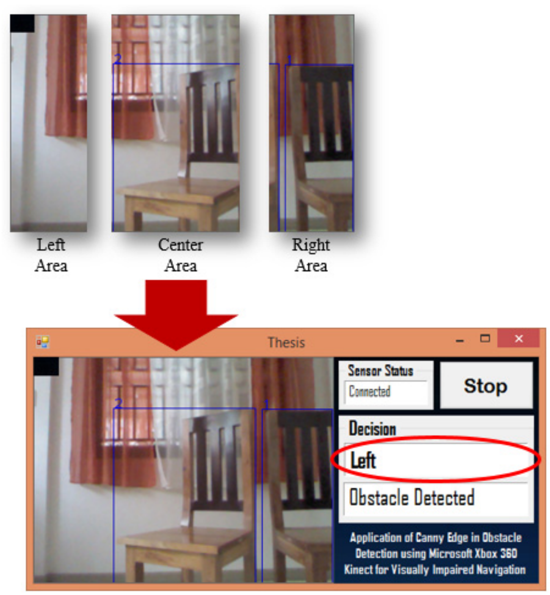

Figure 15. Left decision

A "STOP" command was given if obstacles are present in all areas. Figure 16 shows an example of this scenario.

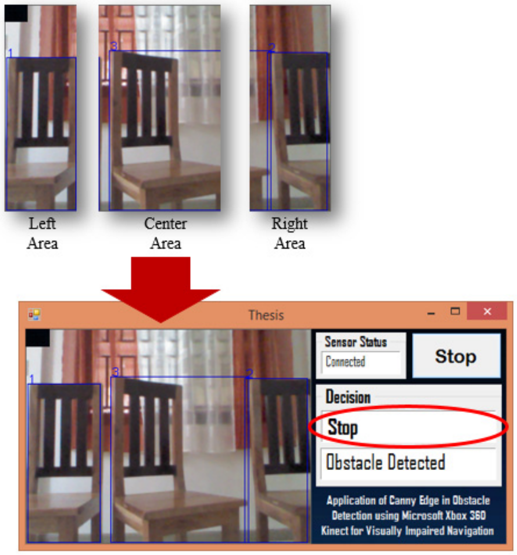

Figure 16. Stop decision

The last block in the diagram is the sound feedback which conveys the instruction through voice synthesis and alerts the user by a beep sound. The system will alert the user through a "beep" sound notification when the detected obstacle reaches 1500 mm and when the distance of the obstacle reaches 1000 mm to 800 mm, the system will give voice notification using voice synthesis.
To further assess the overall performance of the system, the system employed a trial case involving two respondents, a blind-folded and a blind person.

The Kinect sensor was fastened at the user's lower abdominal area. To supply power for the Kinect, a power bank was used which was contained in the backpack together with the laptop that processed the data captured by the Kinect. Also, a headphone had to be worn for the system-to-user communication where notification and command were given.

Using the system, each user navigated in an arranged environment constructed by the authors. The environment was composed of chairs, open door and tables as obstacles. The users were first oriented on how the system works and were given enough time to be familiarized. The map of the obstacle course for the assessment of the system is shown in Figure 17.

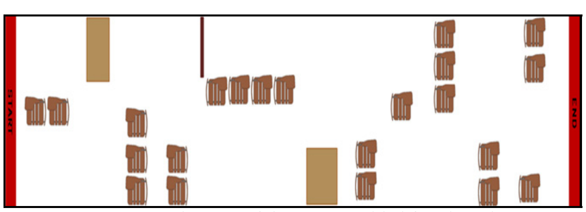

Figure 17. The map of the area provided for obstacles

3. Results

The results of the data gathered in determining the performance of the system in terms of its reliability in the detection of obstacles were shown in this section. The collected samples were composed of 355 images with loose obstacles and 30 images with obstacles with large horizontal frames.
The rate of detection was obtained after carrying out tests on loose obstacles and on obstacles having large surface. The results were specified in Table-2.

Table-2. Results of the Study

	No. of samples	No. of frames with successful detection	No. of frames with failed detection
Loose obstacles	355	334	21
With large horizontal planes	30	30	0
Total	385	364	21
Percentage		**94.5455%**	**5.4545%**

This shows that the system can effectively detect loose obstacles and obstacles with large horizontal planes. The performance of the system is further assessed in a structured environment through a real-time navigation. The test case was conducted by a blindfolded and a blind individual by traversing the same environment. Table-3 shows the difference in the time of navigating between the blindfolded and blind user.

Table-3. The time it takes for each user to navigate the area

User	Duration of successful trial
Blindfolded User	377 seconds
Blind User	271 seconds

The blindfolded user was tasked to navigate using the Kinect-based system in order to sense the environment instead of his own eyes. The vicinity that was used contains the obstacles that were commonly found in an indoor environment such as armchairs, tables and open doors. The successful travel of each user was then recorded and analyzed. Figure 18 shows a person with normal visual sense using the proposed Kinect-based system as a mobility aid in navigating the structured environment arranged by the authors.

Figure-18. The blindfolded person while testing the system

The user was capable of moving in the test environment by just relying on the guidance of the system. But, as expected, the walking speed of the user while using the system is comparatively slower than his normal walking speed. This is because a person with a normal eyesight does not naturally walk as slowly as a blind person walks, and will therefore have difficulty walking at that pace. The user was able to finish the obstacle course successfully without bumping into objects that blocked the pathway. The path taken by the user during the successful travel was presented in Figure 19.

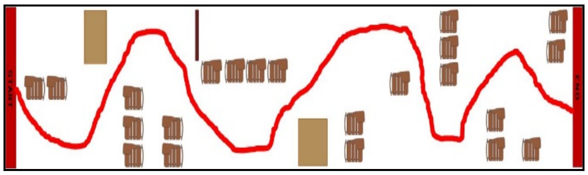

Figure 19. The path taken by the blindfolded user during navigation

In the other trial case, the experiment was participated by a blind person, to know the reliability of the proposed system in detection by providing navigation assistance for a real visually impaired person. Aside from the Kinect-based system, the blind user is not provided any assistive device in the trial navigation. Before navigating on the set upped hallway, the authors let the user be familiarized with how the system works. At first, the blind had difficulties in following the instruction given by the system. But after some trial and familiarization of the system, the user successfully reached the end of the structured environment without colliding with any obstacles in the environment. Figure 20 shows the blind user during the navigation.

Figure 20. The blind person during the system testing

It is observed that the walking speed of the blind user was close to his natural walking pace. However, the proponent also noticed that as the user encounters obstacles, his walking speed decreases compared to the speed on a forward command. The reason for this is the user takes time to respond to the command and change direction in order to find a safe path to take. Figure 21 illustrates the path taken by the blind user during the experiment.

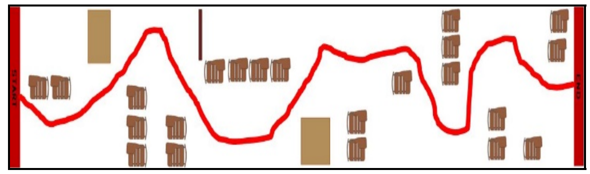

Figure 21. The path taken by the blind user during navigation

The path navigated by both participants can be seen in a voronoi diagram which also shows the alternative routes the users can take while navigating in the middle of the obstacles. Figure 22 shows the obstacle avoiding paths based on a voronoi diagram. The obstacles are represented by a simple four-sided polygon in the diagram.

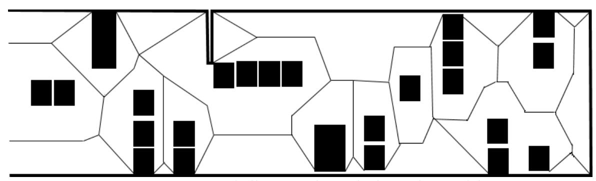

Figure 22. Voronoi diagram based obstacle avoidance

4. Conclusion

The authors were able to perceive obstacles with large horizontal planes and loose obstacles and apply it in the navigation of the visually impaired. Based on the data gathered from the 385 samples, which was composed of 355 samples for loose obstacles and 30 sample frames for obstacles with large horizontal planes, an average success rate of 94.5455% and minimal average failure rate of 5.4545% was obtained. The minimal error rate of detecting the obstacles was often due to the small area of the contour formed. Likewise, the system was also found effective in real-time navigation. Between the blindfolded and blind user, it was found that the blind person better comprehends the structured environment in terms of the duration of their navigation of 377 seconds and 271 seconds, respectively. Hence, both the blind and blind-folded person successfully used the system as they navigate through the structured environment.

5. References

[1] WHO.2014. http://www.who.int/mediacentre/factsheets/

[2] Nawer, A., Hossain, F., Anwar, G. 2015. Ultrasonic Navigation System for the visually impaired & blind pedestrians. American Journal of Engineering Research. 4(2): 13-18.

[3] Harsur, A. and Chitra, M. 2015. Voice Based Navigation System for Blind People Using Ultrasonic Sensor. International Journal on Recent and Innovation Trends in Computing and Communication. 3(6): 4117-4122.

[4] Saputra, M.R.U. and Santosa, P.I. 2014. Obstacle Avoidance for Visually Impaired Using Auto-adaptive Thresholding on Kinect's Depth Image. In Ubiquitous Intelligence and Computing, 2014 IEEE 11th Intl Conf on Autonomic and Trusted Computing and IEEE 14th Intl Conf on Scalable Computing and Communications and Its Associated Symposia/Workshops (UTC-ATC-ScalCom). pp 337-342.

[5] Kanwal, N., Bostanci, E., Currie, K. and Clark, A.F. 2015. A Navigation System for the Visually Impaired: A Fusion of Vision and Depth Sensor. Applied Bionics and Biomechanics.

[6] Pham, H., Le, T., and Vuillerme, N. 2016. Real-Time Obstacle Detection System in Indoor Environment for the Visually Impaired Using Microsoft Kinect Sensor. Journal of Sensors.

[7] Riener, A. and Hartl, H. 2012. Personal Radar: A Self-governed Support System to Enhance Environmental Perception. In Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers. pp 147-156.

[8] Ray, D. 2013. Edge Detection in Digital Image Processing. University of Washington: Department of Mathematics.

[9] Maini, R. and Aggarwal, H. 2009. Study and Comparison of Various Image Edge Detection Techniques. International Journal of Image Processing (IJIP). 3(1): 1-11.

[10] Canny, J. 1986. A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698.

Biographies

Danica C. Barrinuevo is currently taking her degree in Bachelor of Science in Electronics Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2017.

Gladys G. Jacosalem is a graduate of Bachelor of Science in Electronics Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2017.

Jay Neil T. Lagahit is currently taking his degree in Bachelor of Science in Electronics Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2017.

Marcella Joy C. Lobedica is a graduate of Bachelor of Science in Electronics Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2017.

Maria Rebecca T. Salar is currently taking her degree in Bachelor of Science in Electronics Engineering from the Polytechnic University of the Philippines Santa Rosa Campus, Laguna in 2017.

Roselito E. Tolentino is a registered Electronics Engineer and IECEPMember. He is a graduate of Bachelor of Science in Electronics and Communication Engineering at Adamson University in 2004. He finished his Master of Science in Electronics Engineering Major in Control System at Mapua Institute of Technology. He currently takes up Doctor of Philosophy in Electronics Engineering at the same Institute. He is currently working as a part time instructor at PUP - Santa Rosa Campus and DLSU - Dasmarinas.